UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/559,313	03/19/2007	Michael Alex Krieger	680622-2001.1	8888

7590   02/12/2010

John R Lane
Frommer Lawrence & Haug
745 Fifth Avenue
New York, NY 10151

EXAMINER
TRAN, QUOC A

ART UNIT	PAPER NUMBER
2176	

MAIL DATE	DELIVERY MODE
02/12/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on <u>16 October 2009</u>.

2a) ☒ This action is **FINAL**.   2b) ☐ This action is non-final.

3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) <u>20-68</u> is/are pending in the application.

     4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) ☐ Claim(s) _____ is/are allowed.

6) ☒ Claim(s) <u>20-68</u> is/are rejected.

7) ☐ Claim(s) _____ is/are objected to.

8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☒ The specification is objected to by the Examiner.

10) ☒ The drawing(s) filed on <u>16 October 2009</u> is/are: a)☒ accepted or b)☐ objected to by the Examiner.

     Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

     Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

     a)☐ All   b)☐ Some * c)☐ None of:

        1.☐ Certified copies of the priority documents have been received.

        2.☐ Certified copies of the priority documents have been received in Application No. _____.

        3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

     * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
     Paper No(s)/Mail Date <u>10/16/2009</u>.

4) ☐ Interview Summary (PTO-413)
     Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____ .

DETAILED ACTION

This is a Final Office Action in response to Apllicant's responses filed 10/16/2009,

which claims priority of PCT/US04/22200, which claims priority from provisional No.

60/485,472, filed **07/08/2003** (by US LYNX).

- Claims 20-68 are pending.

- Claims 20 and 44 are independent claims.

- Claims 1-19 are CANCELED.

- Claims 20-68 are NEW.

Information Disclosure Statement

A signed and dated copy of applicant's IDS, which were filed on 10/16/2009, is

attached to this Office Action.

It is noted, a portion of the references cited in the Information Disclosure

Statement filed 10/16/2009 (see the NPL section item No. 3) fails to comply with the

provisions of 37 CFR 1.97, 1.98 and MPEP § 609 because one the references does not

list the RELEVANT PAGES [RULE 1.98 (B) (5)] of the publication as required. The

examiner has not considered and has lined through, that portion of the Information

Disclosure Statement as to the merits (see the attachments strike-out line items for

details). Applicant is advised that the date, title and author of any re-submission of any

item of information contained in these Information Disclosure Statements or the

submission of any missing elements will be the date, title and author of submission for

purposes of determining compliance with the requirements based on the time of filing

the statement, and title and author including all certification requirements for statements

under 37 CFR 1.97(e). See MPEP § 609.05(a).

Specification

The specification is objected to as failing to provide proper antecedent basis for

the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction

of the following is required: the recited "*medium*" of Claim 68. The Specification does

not provide support or antecedent basis for the recited "*medium*" that allows the

meaning of the term to be ascertained, as required in 37 CFR 1.75(d)(1).

Claim Objections

Claims 30, 33-36, 38-39, 46, 54, 57-59 and 62-67 are objected to because of the

following informalities:

All paragraphs in the claim, which follow the "…An automatic publishing system

according to claim …, **INCLUDING**…" which is not referring back to and further limiting

to a preceding claims.

In the interest of compact prosecution, the application is further examined against

the prior art, as stated below, upon the assumption of the following:

> ➢ 30, 33-36 and 38-39. (Currently Amended) An automatic publishing
>
>   system according to claim …, [including] <u>further comprising:</u> ….

> ➢ 46, 54, 57-59 and 62-67. (Currently Amended) An automatic publishing
>
>   method according to claim …, [including] <u>further comprising:</u> ….

If the Applicant disagrees with the examiner interpretation, then Applicant should specifically point it out the corrected claim language of the above claims in the response to this Office Action.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 20-43 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 20-43:

In summary, Claim 20 recites " A *document publishing system*" comprising a "*number of components*" having content associated therewith.

The invention recited in Claim 8 is nonstatutory because the "*publishing system*" is software *per se* (see the current disclosure at the Abstract, the first sentence; @ page 14 lines 5-25 and @ page 6 lines 20-25, e.g., a content library, a document type definition; a document manager; a document structure and an output system...);

The recited "*publishing system*" is computer software *per se* in that the "*number of components*" having content associated therewith are simply a computer program. Thus, the recited "*publishing system*" of Claim 20 is not a "process," a "machine," a "manufacture" or a "composition of matter," as defined in 35 U.S.C. 101.

Accordingly, Claim 20 fails to recite statutory subject matter under 35 U.S.C. 101.

Claims 21-43 merely recite either additional components having content

associated there with. Accordingly, Claims 21-43 fail to recite statutory subject matter

under 35 U.S.C. 101.

In the interest of compact prosecution, the application is further examined against

the prior art, as stated below, upon the assumption that the applicants may overcome

the above stated rejections under 35 U.S.C. 101.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> *(a) A patent may not be obtained though the invention is not identically disclosed or
> described as set forth in section 102 of this title, if the differences between the subject
> matter sought to be patented and the prior art are such that the subject matter as a whole
> would have been obvious at the time the invention was made to a person having ordinary
> skill in the art to which said subject matter pertains. Patentability shall not be negatived
> by the manner in which the invention was made.*

Claims 20-68 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Reulein et al., (US 20040205656A1- filed 01/30/2002) [hereinafter "Reulein"], in view of

Davis (US 20090089657A1- Division of No. 09/573,778 filed 05/18/2000) [hereinafter

"Davis"].

Regarding **independent claim 20**, Reulein teaches:

An automated document publishing system,

(At figure 1 and @ the Abstract and at page 2 paragraph 24→ Reulein discloses this

limitation, as clearly indicated in the cited text [e.g., automatic creating and publishing

document system is provided.])
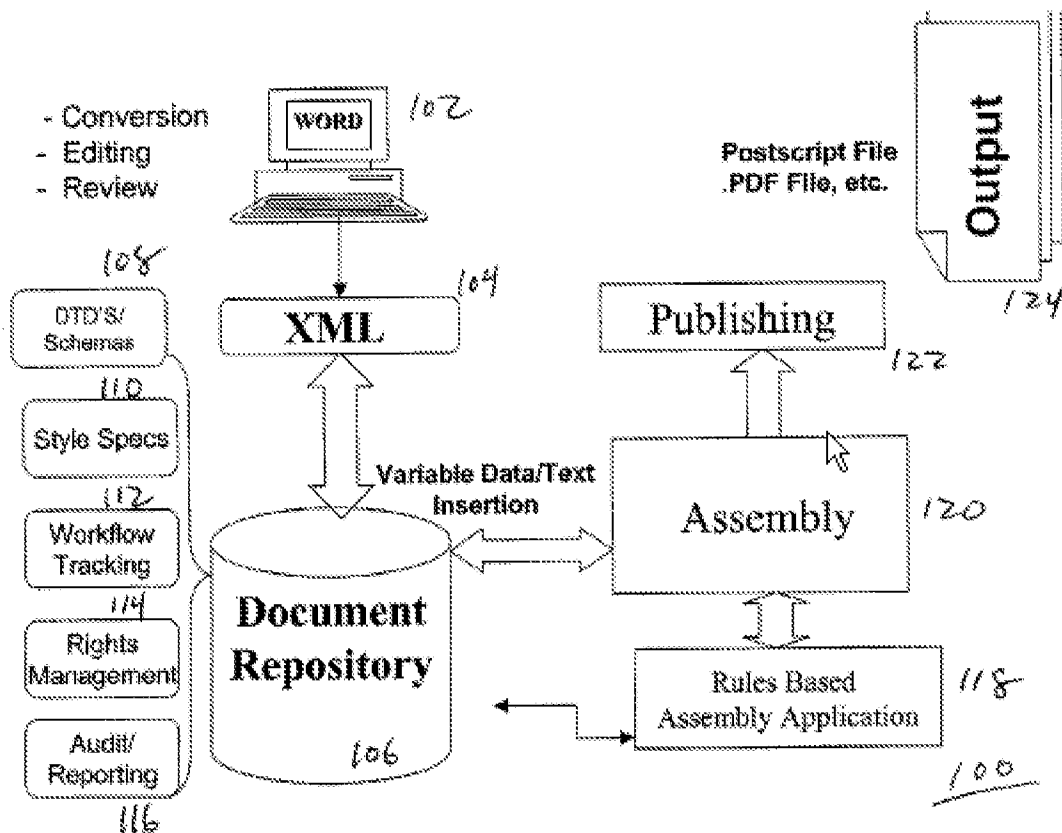
publishing system, comprising: a document type store containing a

plurality of document types,

(At figure 1 and @ Para 23 → Reulein discloses this limitation, as clearly indicated in

the cited text [e.g., Published documents and XML document components are stored in

one or more repositories 106 for subsequent retrieval. The repositories 106 also

maintain document type definitions (DTDs) 108.])

each document type identifying a document structure definition for use in the construction of a document edition, each said element being defined by a said element definition and identifying a said component;

(At Para 28→ Reulein discloses a publishing system includes DTD or schema 108; which is a standard blueprint for the permitted construction of a document. DTDs 108 are used to validate proper assembly and structure of each document prior to publication.

Also Reulein further teaches XML components are mapped to appropriate Document Type Definitions (**DTD**) and/or **schemas** and style specifications which identify any further modifications that may be necessary as a result of text revisions. If any are identified, the document specialist implements such modifications. In step 248 the document, **DTD** and style sheet are loaded into a testing area where the document is tested to verify the document integrity. A "proof" document (such as a PDF file) is returned to a workflow designated user for approval. After the user approves the proof, in step 249 the document specialist checks the XML components into the Master XML Repository and updates the production database. Preferably, document specialists convert word processor files to XML and use an XML editor to modify XML component text and tags [@ Para 43].

Therefore, as broadly disclosed in the instant specification at Page14 lines 1-4, and Lines 13-16 and @ Page 15 Lines 22-33, which is stated "*structure document definition, call* publication type......the last edition...now considered the foundation edition for building the new edition....underneath this whole system lie structure

maps....Theses *structure maps follow a standardized syntax for describing*

documents...and may use XML Document Type Definitions (DTDs) or an XML

schema..", it is reasonable to find that the term, "each document type identifying a

document structure definition " is broad enough to reasonably interprets as XML

components are mapped to appropriate Document Type Definitions (DTD) and/or

schemas and style specifications to output "proof" document (such as a PDF file) in

Reulein. Thus examiner concludes, reasonably, that the claimed, said document

Reulein.

a business data store containing business data; a content library

store containing a library of content components, each component being

capable of use in a plurality of documents;

(At Para 33→ Reulein discloses a publishing system includes the document component

management tool and system of the present invention is multi-faceted, allowing authors

to create a document as well as components for the document. Those document

components are stored in the repository 106 (e.g., a content library store) for document

assembly.)

a document structure store storing at least one said document

structure for at least one respective document edition to be published; and

an output system for forming a structured serial document for publishing

using said document structure.

(At Para 28→ Reulein discloses a publishing system includes DTD or schema 108;

which is a standard blueprint for the permitted construction of a document. DTDs 108

are used to validate proper assembly and structure of each document prior to

publication. Also Reulein further teaches the assembly unit 120 executes the rules

which are loaded into the configuration unit 118 to assemble the document components

and insert and/or remove specific information, such as variable data variable text. The

assembly unit 120 passes a copy of each final document to the publishing unit 122 for

publication. Publishing unit 122 publishes the completed document as a file 124 in a

user specified format (typically an uneditable format) for publication on one or more

output media including printed on paper, fixed on a reusable computer media (e.g.,

CDROM) or as a web viewable file [@ para 24].)

In addition, Reulein does not expressly teach, but Davis teaches:

an element store containing a plurality of elements for use in the

construction of a document edition, each said element identifying a said

component and/or at least one other said element;

(@ Para 301→Davis discloses each element may include a list of attributes and/or an

association with one or more sub elements. Also Davis further teaches each said

element identifying a said component [@ Para 312, e.g., element assigns a default

value, which can be associated with graphical components in the parameters panel in

the Reusable Data Markup Language (hereinafter RDML) data viewer].)

a document manager for using a selected said document type, said

business data, and said elements to form a document structure for a

document edition to identify a plurality of said elements, each said element

identifying a said component and/or at least one other said element; each

document structure definition comprising a hierarchy of element

definitions;

(@ Para 297 and 312→Davis discloses processor 708 receives the parsed text and

creates a tree-shaped data structure of the data elements, matching the structure of the

RDML DTD hierarchy. The hierarchical structure conforms to the XML DOM Model,

meaning that data is available in a standard form, and that a standard library of methods

and functions are available for accessing data, editing elements, searching through

nodes to find certain elements, and so forth (e.g. a document manager for using a

selected said document type, [@ Para 187]. Also Davis further teaches the RDML

Document Type Definition (hereinafter DTD) which describes the structure and

elements of a RDML document ; wherein the DTD 702 used to define RDML data

documents 102 is structured in a hierarchical tree structure of elements. Each element

may include a list of attributes (displayed in Appendix A) and/or an association with one

or more sub-elements. The DTD 702 specifies which attributes are required and which

are optional for any embodiment of the DTD [@ Para 312].)

Therefore, as broadly disclosed in the instant specification at Page13 lines 32-35,

which is stated "Both elements and components have attributes and properties

(respectively) containing values that may be mapped to specific publication/active

edition properties according to a client's specific business rules", it is reasonable to find

that the term, " *each said element identifying a said component and/or at least one other*

said element " is broad enough to reasonably interprets as "element identifying a said

component" in Davis. Thus examiner concludes, reasonably, that the claimed, said

document *each said element identifying a said component and/or at least one other said*

element is described by Toepfer.

Accordingly, It would have been obvious to one of ordinary skill in the art at the

time of the invention to modify Reulein's publishing documents and XML document

components to include a means of said an element store containing a plurality of

elements for use in the construction of a document edition, each said element

identifying a said component and/or at least one other said element; and a document

manager for using a selected said document type, said business data, and said

elements to form a document structure for a document edition to identify a plurality of

said elements, each said element identifying a said component and/or at least one other

said element; and each document structure definition comprising a hierarchy of element

definitions as taught by Davis; because they are both from the analogous art of markup

document publishing system. Therefore, the artisan would have well appreciated that

Davis relates to general method of publishing documents and XML document

components of Reulein. This is done in an iterative manner that allows publishing unit

publishes the completed document as a file in a user specified format (typically an

uneditable format) for publication on one or more output media including printed on

paper, fixed on a reusable computer media and may be published on output media,

e.g., printed by a local print facility on a PC as a PDF and/or in hardcopy by a

connected [@ para 23-26 and 32 of Reulein].)

Regarding **independent claim 44:**

the rejection of claim 20 is fully incorporated. Thus, Reulein and Davis

disclose every limitation of Claim 44 and provide proper reasons to combine as

indicated in the above rejections for Claim 20.

Claim 21,

Reulein teaches the method of claim 20 and further comprises:

wherein said element definitions define rules and attributes, said

elements include rules related to said business data and attributes and

said document manager is adapted to form said document structure using

said attributes and rules.

(@ Para 130→Davis discloses elements of the RDML Document Type Definition 702. In

one implementation consistent with the present invention, RDML documents 102

conform to the rules provided by the DTD 702 (also shown on FIG. 7). In accordance

with one implementation of the present invention, an RDML DTD 702 is shown at

Appendix A. Attributes and elements of the DTD 702 may also be seen in the full

sample RDML document 102 in Appendix B. Those two Appendices A and B are useful

for examining specific attributes and elements of the DTD.

Accordingly, It would have been obvious to one of ordinary skill in the art at the

time of the invention to modify Reulein's publishing documents and XML document

components to include a means of said element definitions define rules and attributes,

said elements include rules related to said business data and attributes and said

document manager is adapted to form said document structure using said attributes and

rules as taught by Davis; because they are both from the analogous art of markup

document publishing system. Therefore, the artisan would have well appreciated that

Davis relates to general method of publishing documents and XML document

components of Reulein. This is done in an iterative manner that allows publishing unit

publishes the completed document as a file in a user specified format (typically an

uneditable format) for publication on one or more output media including printed on

paper, fixed on a reusable computer media and may be published on output media,

e.g., printed by a local print facility on a PC as a PDF and/or in hardcopy by a

connected [@ para 23-26 and 32 of Reulein].)

Claim 22,

Reulein teaches the method of claim 20 and further comprises:

wherein said document manager is adapted to autopopulate said

document structure based on said selected document type.

(At figure 1 and @ the Abstract and at page 2 paragraph 24→ Reulein discloses this

limitation, as clearly indicated in the cited text [e.g., automatic creating and publishing

document system is provided.] Also Reulein further teaches XML components are

mapped to appropriate Document Type Definitions (**DTD**) and/or **schemas** and style

specifications [@ Para 43].

Claim 23,

> Reulein teaches the method of claim 22 and further comprises:

> > **wherein said document manager is adapted to autopopulate said**

> > **document structure by selecting and evaluating candidate elements**

> > **against said business data.**

(At figure 1 and @ the Abstract and at page 2 paragraph 24→ Reulein discloses this

limitation, as clearly indicated in the cited text [e.g., automatic creating and publishing

document system is provided.] Also Reulein further teaches XML components are

mapped to appropriate Document Type Definitions (**DTD**) and/or **schemas** and style

specifications based upon business rules [@ Para 43 and 53-54].

Claim 24,

> Reulein teaches the method of claim 22 and further comprises:

> > **wherein said document manager is adapted to autopopulate said**

> > **document structure using a said document structure of a previous edition**

> > **and re-evaluating the elements and identified components against said**

> > **business data.**

(At figure 1 and @ the Abstract and at page 2 paragraph 24→ Reulein discloses this

limitation, as clearly indicated in the cited text [e.g., automatic creating and publishing

document system is provided.] Also Reulein further teaches XML components are

mapped to appropriate Document Type Definitions (**DTD**) and/or **schemas** and style

specifications based upon business rules [@ Para 43 and 53-54]; Also see Reulein @

Para 33, discloses document components are stored in the repository 106 for recall

and re-edit a previously stored document components.)

. *Claim 25*,

  Reulein teaches the method of claim 22 and further comprises:

    wherein said document manager is adapted to autopopulate said

document structure using rules based on contexts and attributes for said

elements and properties of identified components and said business data.

(At figure 1 and @ the Abstract and at page 2 paragraph 24→ Reulein discloses this

limitation, as clearly indicated in the cited text [e.g., automatic creating and publishing

document system is provided.] Also Reulein further teaches XML components are

mapped to appropriate Document Type Definitions (**DTD**) and/or **schemas** and style

specifications (context attributes) based upon business rules [@ Para 43 and 53-54].

Also see Para 47, Reulein further teaches the assembly subsystem 166 assembles

document components according to predefined assembly rules and inserts or deletes

the unique variable text and, then, inserts data from variable data files in response to

tagged attributes of the specified XML components.).

Claim 26:

Reulein teaches the method of claim 24 and further comprises:

wherein said document manager is adapted to autopopulate said

document structure using rules based on contexts and attributes for said

elements and properties of identified components, said business data and

dynamic substitution of components.

(@ Para 130→Davis discloses elements of the RDML Document Type Definition 702. In

one implementation consistent with the present invention, RDML documents 102

conform to the rules provided by the DTD 702 (also shown on FIG. 7). In accordance

with one implementation of the present invention, an RDML DTD 702 is shown at

Appendix A. Attributes and elements of the DTD 702 may also be seen in the full

sample RDML document 102 in Appendix B. Those two Appendices A and B are useful

for examining specific attributes and elements of the DTD.)

Accordingly, It would have been obvious to one of ordinary skill in the art at the

time of the invention to modify Reulein's publishing documents and XML document

components to include a means of said document manager is adapted to autopopulate

said document structure using rules based on contexts and attributes for said elements

and properties of identified components, said business data and dynamic substitution of

components as taught by Davis; because they are both from the analogous art of

markup document publishing system. Therefore, the artisan would have well

appreciated that Davis relates to general method of publishing documents and XML

document components of Reulein. This is done in an iterative manner that allows

publishing unit publishes the completed document as a file in a user specified format

(typically an uneditable format) for publication on one or more output media including

printed on paper, fixed on a reusable computer media and may be published on output

media, e.g., printed by a local print facility on a PC as a PDF and/or in hardcopy by a

connected [@ para 23-26 and 32 of Reulein].)

. *Claim 27*,

> Reulein teaches the method of claim 20 and further comprises:

> > **wherein each said document structure definition has a root element**

> > **defining only other elements.**

(@ Para 297 and 312→Davis discloses processor 708 receives the parsed text and

creates a tree-shaped data structure of the data elements, matching the structure of the

RDML DTD hierarchy.

examining specific attributes and elements of the DTD.)

> Accordingly, It would have been obvious to one of ordinary skill in the art at the

time of the invention to modify Reulein's publishing documents and XML document

components to include a means of said document structure definition has a root

element defining only other elements as taught by Davis; because they are both from

the analogous art of markup document publishing system. Therefore, the artisan would

have well appreciated that Davis relates to general method of publishing documents

and XML document components of Reulein. This is done in an iterative manner that

allows publishing unit publishes the completed document as a file in a user specified

format (typically an uneditable format) for publication on one or more output media

including printed on paper, fixed on a reusable computer media and may be published

on output media, e.g., printed by a local print facility on a PC as a PDF and/or in

hardcopy by a connected [@ para 23-26 and 32 of Reulein].)

. *Claim 28*,

Reulein teaches the method of claim 20 and further comprises:

wherein said document edition includes mandated content required

to be present in the document edition, said document manager is adapted

to use said document type, said business data, and said elements to

identify mandated elements that are required to be present in the document

structure to identify required components, to include said required

elements in the document structure and to identify in the document

structure any said required elements missing in said element store, said

any missing elements identifying any missing components in said content

library.

(@ Para 297 and 312→Davis discloses processor 708 receives the parsed text and

creates a tree-shaped data structure of the data elements, matching the structure of the

RDML DTD hierarchy. The hierarchical structure conforms to the XML DOM Model,

meaning that data is available in a standard form, and that a standard library of methods

and functions are available for accessing data, *editing elements*, searching through

nodes to find certain elements, and so forth (e.g. a document manager for using a

selected said document type, [@ Para 187]. Also Davis further teaches the RDML

Document Type Definition (hereinafter DTD) which describes the structure and

elements of a RDML document ; wherein the DTD 702 used to define RDML data

documents 102 is structured in a hierarchical tree structure of elements. Each element

may include a list of attributes (displayed in Appendix A) and/or an association with one

or more sub-elements. The DTD 702 specifies which attributes are required and which

are optional for any embodiment of the DTD [@ Para 312]. Also Davis further discloses

RMML builds error handling into its interpreter and makes available automated testing

tools to handle error of programming language dependencies, data type inconsistencies

and so on [@ Para 84, 90, 118 and 188].)

Accordingly, It would have been obvious to one of ordinary skill in the art at the

time of the invention to modify Reulein's publishing documents and XML document

components to include a means of said document edition includes mandated content

required to be present in the document edition, said document manager is adapted to

use said document type, said business data, and said elements to identify mandated

elements that are required to be present in the document structure to identify required

components, to include said required elements in the document structure and to identify

in the document structure any said required elements missing in said element store,

said any missing elements identifying any missing components in said content library as

taught by Davis; because they are both from the analogous art of markup document

publishing system. Therefore, the artisan would have well appreciated that Davis relates

to general method of publishing documents and XML document components of Reulein.

This is done in an iterative manner that allows publishing unit publishes the completed

document as a file in a user specified format (typically an uneditable format) for

publication on one or more output media including printed on paper, fixed on a reusable

computer media and may be published on output media, e.g., printed by a local print

facility on a PC as a PDF and/or in hardcopy by a connected [@ para 23-26 and 32 of

Reulein].)

Claim 29,

> Reulein teaches the method of claim 28 and further comprises:

> > **wherein said output system is adapted to only publish document**

> > **editions having all required elements and identified components.**

(At Para 28→ Reulein discloses a publishing system includes DTD or schema 108;

which is a standard blueprint for the permitted construction of a document. DTDs 108

are used to validate proper assembly and structure of each document prior to

publication. Also Reulein further teaches the assembly unit 120 executes the rules

which are loaded into the configuration unit 118 to assemble the document components

and insert and/or remove specific information, such as variable data variable text. The

assembly unit 120 passes a copy of each final document to the publishing unit 122 for

publication. Publishing unit 122 publishes the completed document as a file 124 in a

user specified format (typically an uneditable format) for publication on one or more

output media including printed on paper, fixed on a reusable computer media (e.g.,

CDROM) or as a web viewable file [@ para 24].)

Claim 30,

>   Reulein teaches the method of claim 20 and further comprises:

>> **including an archive store for storing said structured serial**

>> **document and the associated document structure for published**

>> **documents.**

(At figure 1 and @ Para 23 → Reulein discloses this limitation, as clearly indicated in

the cited text [e.g., Published documents and XML document components are stored in

one or more repositories 106 for subsequent retrieval. The repositories 106 also

maintain document type definitions (DTDs) 108.])

Claim 31,

>   Reulein teaches the method of claim 20 and further comprises:

>> **wherein said components in said content library are stored as**

>> **groups of components having defined relationship properties, wherein said**

>> **document manager is adapted to select a said component of a said group**

>> **in said document structure using said document type, said business data**

>> **and said elements.**

(At figure 1 and @ Para 23 → Reulein discloses the published documents and XML

document components are stored in one or more repositories 106 for subsequent

retrieval. The repositories 106 also maintain document type definitions (DTDs) 108.])

Claim 32,

Reulein teaches the method of claim 31 and further comprises:

wherein said components of a group comprise text in different

languages, said publication type defines a publication language and said

document manager is adapted to form a document structure identifying

language components based on the language defined in said selected

document type.

(At figure 1 and @ Para 23 → Reulein discloses the published documents and XML

document components are stored in one or more repositories 106 for subsequent

retrieval. The repositories 106 also maintain document type definitions (DTDs) 108.]

Also Davis further discloses Character_set designates which standard character set is

represented, thereby allowing for support for **foreign languages** [@ Para 145].).

Claim 33,

Reulein teaches the method of claim 32 and further comprises:

including a translation system for identifying that a component in a group requires translation and notifying a translator that said component requires translation to create a new component in the group.

(At figure 1 and @ Para 23 → Reulein discloses the published documents and XML document components are stored in one or more repositories 106 for subsequent retrieval. The repositories 106 also maintain document type definitions (DTDs) 108.] Also Davis further discloses Character_set designates which standard character set is represented, thereby allowing for support for **foreign languages and translator** [@ Para 145 and @ Para 159].).

Claim 34,

Reulein teaches the method of claim 20 and further comprises:

including a component interface to allow users to edit and/or add to said components in said content library.

(@ Para 43 → Reulein discloses an XML editor to modify XML component text and tags.).

Claim 35,

Reulein teaches the method of claim 20 and further comprises:

wherein said proof generator is adapted to generate a proof by comparison to a previously generated proof.

(@ Para 43 → Reulein discloses a "proof" document (such as a PDF file) is returned to a workflow designated user for approval. After the user approves the proof, in step 249 the document specialist checks the XML components into the Master XML Repository and updates the production database).

Claim 36,

Reulein teaches the method of claim 35 and further comprises:

including a proof archive for storing the static representation of a proof and the associated document structure.

(@ Para 43 → Reulein discloses a "proof" document (such as a PDF file) is returned to a workflow designated user for approval. After the user approves the proof, in step 249 the document specialist checks the XML components into the Master XML Repository and updates the production database [a proof archive]).

Claim 37,

Reulein teaches the method of claim 36 and further comprises:

wherein said proof generator is adapted to generate a proof by comparison to a previously generated proof.

(@ Para 43 → Reulein discloses a "proof" document (such as a PDF file) is returned to a workflow designated user for approval. After the user approves the proof, in step 249 the document specialist checks the XML components into the Master XML Repository and updates the production database [a proof archive]).

Claim 38,

Reulein teaches the method of claim 20 and further comprises:

including a user interface for editing a document structure for a document edition, the user interface including first interface means for generating an image of AT LEAST ONE component in a first display region and enabling a user to edit said AT LEAST ONE component; and second interface means for generating an interface in a second display region to allow a user to make structural changes to the document structure, said second interface means being adapted to display handles in said second display region, each handle comprising an iconized representation of an element to allow selection and manipulation of the component identified by the element and being displayed in said second display region at a position adjacent to the component displayed in said first display region, and said handles being displayed as an organizational structure defining the structure of the elements of the document edition.

(@ Para 297 and 312 and illustrates in figure 14A-F→Davis discloses processor 708 receives the parsed text and creates a tree-shaped data structure of the data elements, matching the structure of the RDML DTD hierarchy. The hierarchical structure conforms to the XML DOM Model, meaning that data is available in a standard form, and that a standard library of methods and functions are available for accessing data, ***editing elements***, searching through nodes to find certain elements, and so forth (e.g. a document manager for using a selected said document type, [@ Para 187]. Also Davis

further teaches the RDML Document Type Definition (hereinafter DTD) which describes the structure and elements of a RDML document ; wherein the DTD 702 used to define RDML data documents 102 is structured in a hierarchical tree structure of elements. Each element may include a list of attributes (displayed in Appendix A) and/or an association with one or more sub-elements. The DTD 702 specifies which attributes are required and which are optional for any embodiment of the DTD [@ Para 312]. Also Davis shown in FIGS. 14A-D, 16 and 17, the graphical user interfaces ("GUI") 734 and HTML Browser, which has several responsibilities. Generally, it creates itself and other visual components upon start-up of the application, and provides a central storage place for a minimal number of global variables of the application (such as file directories, etc.). Further, it responds to user actions, such as mouse clicks and keyboard shortcuts, and repaints the screen, or portions of the screen, at appropriate moments [@ Para 297-298]. Also @ Para 172 and 186, Davis discloses the use of "icon" element provides a place for giving the unit an icon to be used in lists, drop-down boxes, etc.)

Accordingly, It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Reulein's publishing documents and XML document components to include a means of said including a user interface for editing a document structure for a document edition, the user interface including first interface generating an image of component in a first display region and enabling a user to edit said component; and second interface generating an interface in a second display region to allow a user to make structural changes to the document structure, said second interface being

adapted to display handles in said second display region, each handle comprising an

iconized representation of an element to allow selection and manipulation of the

component identified by the element and being displayed in said second display region

at a position adjacent to the component displayed in said first display region, and said

handles being displayed as an organizational structure defining the structure of the

elements of the document edition as taught by Davis; because they are both from the

analogous art of markup document publishing system. Therefore, the artisan would

have well appreciated that Davis relates to general method of publishing documents

and XML document components of Reulein. This is done in an iterative manner that

allows publishing unit publishes the completed document as a file in a user specified

format (typically an uneditable format) for publication on one or more output media

including printed on paper, fixed on a reusable computer media and may be published

on output media, e.g., printed by a local print facility on a PC as a PDF and/or in

hardcopy by a connected [@ para 23-26 and 32 of Reulein].)

Claim 39,

Reulein teaches the method of claim 38 and further comprises:

including third interface means for generating an image in a third

display region of a tree structure representing the structure of a plurality of

said elements defining said document structure, wherein each element

identifies a said component and/or at least one other element, and said

third interface means is adapted to enable a user to select components to

be displayed in said first display region.

(@ Para 297 and 312 and illustrates in figure 14A-F→Davis discloses processor 708

receives the parsed text and creates a tree-shaped data structure of the data elements,

matching the structure of the RDML DTD hierarchy. The hierarchical structure conforms

to the XML DOM Model, meaning that data is available in a standard form, and that a

standard library of methods and functions are available for accessing data, *editing*

elements, searching through nodes to find certain elements, and so forth (e.g. a

document manager for using a selected said document type, [@ Para 187]. Also Davis

further teaches the RDML Document Type Definition (hereinafter DTD) which describes

the structure and elements of a RDML document ; wherein the DTD 702 used to define

RDML data documents 102 is structured in a hierarchical tree structure of elements.

Each element may include a list of attributes (displayed in Appendix A) and/or an

association with one or more sub-elements (. The DTD 702 specifies which attributes

are required and which are optional for any embodiment of the DTD [@ Para 312]. Also

Davis shown in FIGS. 14A-D, 16 and 17, the graphical user interfaces ("GUI") 734 and

HTML Browser, which has several responsibilities. Generally, it creates itself and other

visual components upon start-up of the application, and provides a central storage place

for a minimal number of global variables of the application (such as file directories, etc.).

Further, it responds to user actions, such as mouse clicks and keyboard shortcuts, and

repaints the screen, or portions of the screen, at appropriate moments [@ Para 297-

298]. Also @ Para 172 and 186, Davis discloses the use of "icon" element provides a

place for giving the unit an icon to be used in lists, drop-down boxes, etc.)

Accordingly, It would have been obvious to one of ordinary skill in the art at the

time of the invention to modify Reulein's publishing documents and XML document

components to include a means of said including third interface means for generating

an image in a third display region of a tree structure representing the structure of a

plurality of said elements defining said document structure, wherein each element

identifies a said component and/or other element, and said third interface means is

adapted to enable a user to select components to be displayed in said first display

region as taught by Davis; because they are both from the analogous art of markup

document publishing system. Therefore, the artisan would have well appreciated that

Davis relates to general method of publishing documents and XML document

components of Reulein. This is done in an iterative manner that allows publishing unit

publishes the completed document as a file in a user specified format (typically an

uneditable format) for publication on one or more output media including printed on

paper, fixed on a reusable computer media and may be published on output media,

e.g., printed by a local print facility on a PC as a PDF and/or in hardcopy by a

connected [@ para 23-26 and 32 of Reulein].)

Claim 40,

Reulein teaches the method of claim 39 and further comprises:

wherein said third interface means is adapted to highlight at least

one position in the tree structure representing a position in the tree

structure of at least one element identifying said at least one component

displayed in said first display region.

(@ Para 297 and 312 and illustrates in figure 14A-F→Davis discloses processor 708

receives the parsed text and creates a tree-shaped data structure of the data elements,

matching the structure of the RDML DTD hierarchy. The hierarchical structure conforms

to the XML DOM Model, meaning that data is available in a standard form, and that a

standard library of methods and functions are available for accessing data, *editing*

elements, searching through nodes to find certain elements, and so forth (e.g. a

document manager for using a selected said document type, [@ Para 187]. Also Davis

further teaches the RDML Document Type Definition (hereinafter DTD) which describes

the structure and elements of a RDML document ; wherein the DTD 702 used to define

RDML data documents 102 is structured in a hierarchical tree structure of elements.

Each element may include a list of attributes (displayed in Appendix A) and/or an

association with one or more sub-elements (. The DTD 702 specifies which attributes

are required and which are optional for any embodiment of the DTD [@ Para 312]. Also

Davis shown in FIGS. 14A-D, 16 and 17, the graphical user interfaces ("GUI") 734 and

HTML Browser, which has several responsibilities. Generally, it creates itself and other

visual components upon start-up of the application, and provides a central storage place

for a minimal number of global variables of the application (such as file directories, etc.).

Further, it responds to user actions, such as mouse clicks and keyboard shortcuts, and

repaints the screen, or portions of the screen, at appropriate moments [@ Para 297-298]. Also @ Para 172 and 186 and 90 and 41, Davis discloses the use of "icon" element provides a place for giving the unit an icon to be used in lists, drop-down boxes, etc. and highlighted, underlined text)

Accordingly, It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Reulein's publishing documents and XML document components to include a means of said wherein said third interface means is adapted to highlight a position in the tree structure representing a position in the tree structure of at least one element identifying said component displayed in said first display region as taught by Davis; because they are both from the analogous art of markup document publishing system. Therefore, the artisan would have well appreciated that Davis relates to general method of publishing documents and XML document components of Reulein. This is done in an iterative manner that allows publishing unit publishes the completed document as a file in a user specified format (typically an uneditable format) for publication on one or more output media including printed on paper, fixed on a reusable computer media and may be published on output media, e.g., printed by a local print facility on a PC as a PDF and/or in hardcopy by a connected [@ para 23-26 and 32 of Reulein].)

Claim 41,

Reulein teaches the method of claim 38 and further comprises:

wherein said second interface means is adapted to highlight at least

one position in the structure representing a position in the structure of at

least one element identifying said at least one component displayed in said

first display region.

(@ Para 297 and 312 and illustrates in figure 14A-F→Davis discloses processor 708

receives the parsed text and creates a tree-shaped data structure of the data elements,

matching the structure of the RDML DTD hierarchy. The hierarchical structure conforms

to the XML DOM Model, meaning that data is available in a standard form, and that a

standard library of methods and functions are available for accessing data, *editing*

elements, searching through nodes to find certain elements, and so forth (e.g. a

document manager for using a selected said document type, [@ Para 187]. Also Davis

further teaches the RDML Document Type Definition (hereinafter DTD) which describes

the structure and elements of a RDML document ; wherein the DTD 702 used to define

RDML data documents 102 is structured in a hierarchical tree structure of elements.

Each element may include a list of attributes (displayed in Appendix A) and/or an

association with one or more sub-elements (. The DTD 702 specifies which attributes

are required and which are optional for any embodiment of the DTD [@ Para 312]. Also

Davis shown in FIGS. 14A-D, 16 and 17, the graphical user interfaces ("GUI") 734 and

HTML Browser, which has several responsibilities. Generally, it creates itself and other

visual components upon start-up of the application, and provides a central storage place

for a minimal number of global variables of the application (such as file directories, etc.).

Further, it responds to user actions, such as mouse clicks and keyboard shortcuts, and

repaints the screen, or portions of the screen, at appropriate moments [@ Para 297-298]. Also @ Para 172 and 186 and 90 and 41, Davis discloses the use of "icon" element provides a place for giving the unit an icon to be used in lists, drop-down boxes, etc. and highlighted, underlined text)

Accordingly, It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Reulein's publishing documents and XML document components to include a means of said second interface means is adapted to highlight one position in the structure representing a position in the structure of one element identifying said one component displayed in said first display region as taught by Davis; because they are both from the analogous art of markup document publishing system. Therefore, the artisan would have well appreciated that Davis relates to general method of publishing documents and XML document components of Reulein. This is done in an iterative manner that allows publishing unit publishes the completed document as a file in a user specified format (typically an uneditable format) for publication on one or more output media including printed on paper, fixed on a reusable computer media and may be published on output media, e.g., printed by a local print facility on a PC as a PDF and/or in hardcopy by a connected [@ para 23-26 and 32 of Reulein].)

Claim 42,

Reulein teaches the method of claim 39 and further comprises:

> **wherein said third interface means is adapted to indicate a position**
>
> **in the tree structure of an element identifying a missing component**
>
> **required in the document edition.**

(@ Para 297 and 312 and illustrates in figure 14A-F→Davis discloses processor 708

receives the parsed text and creates a tree-shaped data structure of the data elements,

matching the structure of the RDML DTD hierarchy. The hierarchical structure conforms

to the XML DOM Model, meaning that data is available in a standard form, and that a

standard library of methods and functions are available for accessing data, *editing*

elements, searching through nodes to find certain elements, and so forth (e.g. a

document manager for using a selected said document type, [@ Para 187]). Also Davis

shown in FIGS. 14A-D, 16 and 17, the graphical user interfaces ("GUI") 734 and HTML

Browser, which has several responsibilities. Generally, it creates itself and other visual

components upon start-up of the application, and provides a central storage place for a

minimal number of global variables of the application (such as file directories, etc.).

Further, it responds to user actions, such as mouse clicks and keyboard shortcuts, and

repaints the screen, or portions of the screen, at appropriate moments [@ Para 297-

298]. Also Davis further discloses RMML builds error handling into its interpreter and

makes available automated testing tools to handle error of programming language

dependencies, data type inconsistencies and so on [@ Para 84, 90, 118 and 188].)

Accordingly, It would have been obvious to one of ordinary skill in the art at the

time of the invention to modify Reulein's publishing documents and XML document

components to include a means of said third interface means is adapted to indicate a

position in the tree structure of an element identifying a missing component required in

the document edition as taught by Davis; because they are both from the analogous art

of markup document publishing system. Therefore, the artisan would have well

appreciated that Davis relates to general method of publishing documents and XML

document components of Reulein. This is done in an iterative manner that allows

publishing unit publishes the completed document as a file in a user specified format

(typically an uneditable format) for publication on one or more output media including

printed on paper, fixed on a reusable computer media and may be published on output

media, e.g., printed by a local print facility on a PC as a PDF and/or in hardcopy by a

connected [@ para 23-26 and 32 of Reulein].)

Claim 43,

       Reulein teaches the method of claim 38 and further comprises:

              wherein said second interface means is adapted to indicate a

position in the structure of an element identifying a missing component

required in the document edition.

(@ Para 297 and 312 and illustrates in figure 14A-F→Davis discloses processor 708

receives the parsed text and creates a tree-shaped data structure of the data elements,

matching the structure of the RDML DTD hierarchy. The hierarchical structure conforms

to the XML DOM Model, meaning that data is available in a standard form, and that a

standard library of methods and functions are available for accessing data, *editing*

elements, searching through nodes to find certain elements, and so forth (e.g. a document manager for using a selected said document type, [@ Para 187]). Also Davis shown in FIGS. 14A-D, 16 and 17, the graphical user interfaces ("GUI") 734 and HTML Browser, which has several responsibilities. Generally, it creates itself and other visual components upon start-up of the application, and provides a central storage place for a minimal number of global variables of the application (such as file directories, etc.). Further, it responds to user actions, such as mouse clicks and keyboard shortcuts, and repaints the screen, or portions of the screen, at appropriate moments [@ Para 297-298]. Also Davis further discloses RMML builds error handling into its interpreter and makes available automated testing tools to handle error of programming language dependencies, data type inconsistencies and so on [@ Para 84, 90, 118 and 188].)

Accordingly, It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Reulein's publishing documents and XML document components to include a means of said second interface means is adapted to indicate a position in the structure of an element identifying a missing component required in the document edition as taught by Davis; because they are both from the analogous art of markup document publishing system. Therefore, the artisan would have well appreciated that Davis relates to general method of publishing documents and XML document components of Reulein. This is done in an iterative manner that allows publishing unit publishes the completed document as a file in a user specified format (typically an uneditable format) for publication on one or more output media including

printed on paper, fixed on a reusable computer media and may be published on output

media, e.g., printed by a local print facility on a PC as a PDF and/or in hardcopy by a

connected [@ para 23-26 and 32 of Reulein].)

Claims 45-67: respectively

the rejection of claims 45-67 are fully incorporated. Thus, Reulein and

Davis disclose every limitation of Claims 45-67 and provide proper reasons to

combine as indicated in the above rejections for Claims 21-43.

Claim 68,

Claim 68 is directed toward a computer usable medium implement a

method recited in Claim 44. Thus, Reulein and Davis disclose every limitation of

Claims 45-67 and provide proper reasons to combine as indicated in the above

rejections for Claim 44 (see also @ Para 24, Reulein discloses Publishing unit

includes output media including printed on paper, fixed on a reusable computer

media (e.g., CDROM) or as a web viewable file.)

It is noted that any citations to specific, pages, columns, lines, or figures in the

prior art references and any interpretation of the references should not be considered to

be limiting in any way. A reference is relevant for all it contains and may be relied upon

for all that it would have reasonably suggested to one having ordinary skill in the art.

See, MPEP 2123.

Response to Arguments

Applicant's remarks filed 10/16/2009 have been considered but are moot in view of the new ground(s) of rejection.

It is noted; Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action (see above for details).

Further, the Examiner maintains Reulein reference. Because, Reulein teaches a data structure and method of automatically creating a plurality of individually customized documents; wherein document components are created and stored. Also document condition and action rules governing document assembly; and are created and stored. Document specific data and text specific to individuals is provided with a document request that initiates assembly of a document. Each requested document is assembled automatically in response to the rules and provided data and text [@ Para 9 and illustrated in Fig. 1]. Also Reulein further teaches XML components are mapped to appropriate Document Type Definitions (**DTD**) and/or **schemas** and style specifications which identify any further modifications that may be necessary as a result of text revisions. If any are identified, the document specialist implements such modifications. In step 248 the document, **DTD** and style sheet are loaded into a testing area where the document is tested to verify the document integrity. A "proof" document (such as a PDF file) is returned to a workflow designated user for approval. After the user approves the proof, in step 249 the document specialist checks the XML components into the Master

XML Repository and updates the production database. Preferably, document specialists

convert word processor files to XML and use an XML editor to modify XML component

text and tags [@ Para 43].

Therefore, as broadly disclosed in the instant specification at Page14 lines 1-4,

and Lines 13-16 and @ Page 15 Lines 22-33, which is stated "**structure document**

definition, call publication type.......the last edition...now considered the foundation

edition for building the new edition....underneath this whole system lie structure

maps....Theses **structure maps follow a standardized syntax for describing**

documents...and may use XML Document Type Definitions (DTDs) or an XML

schema..", it is reasonable to find that the term, "each document type identifying a

document structure definition " is broad enough to reasonably interprets as XML

components are mapped to appropriate Document Type Definitions (DTD) and/or

schemas and style specifications to output "proof" document (such as a PDF file) in

Reulein. Thus examiner concludes, reasonably, that the claimed, said document

Reulein.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in

this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP

§ 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37

CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not

mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any

extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later

than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Quoc A. Tran whose telephone number is 571-272-

8664. The examiner can normally be reached on Mon through Fri 8AM - 5PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Doug Hutton can be reached on (571)272-4137. The fax phone number for

the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the

Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR.

Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should

you have questions on access to the Private PAIR system, contact the Electronic

Business Center (EBC) at 866-217-9197 (toll-free).

/Quoc A, Tran/
Patent Examiner

/DOUG HUTTON/
Supervisory Patent Examiner, Art Unit 2176